

2. Годин В.В., Корнеев И.К., «Информационное обеспечение управленческой деятельности» Учебник. – М.:Мастерство; Высшая школа, 2001.-240 с ISBN 5-294-00042-3
3. Единый квалификационный справочник должностей работников образования.

Лившиц А.Л.

МЕТРИКИ СЛОЖНОСТИ ИНФОРМАЦИОННЫХ СИСТЕМ УПРАВЛЕНИЯ ВУЗОМ

slivshits@yandex.ru

*ГОУ ВПО «Уральский государственный технический университет – УПИ имени первого Президента России Б.Н.Ельцина»
г. Екатеринбург*

Работу современного вуза невозможно представить без использования разнообразных информационных систем. Однако использование большого количества разных информационных систем разработанных в разное время и на разных платформах создает существенные сложности в их поддержке и интеграции между собой. По этому все чаще происходит переход от «ласкутной» автоматизации к внедрению единых систем управления вузом.

Так как единые системы управления вузом, это очень большие и сложные система, то важными параметрами, влияющими на успех их разработки и внедрения, являются качество и надежность, трудоемкость разработки и внедрения.

В отличие от большинства отраслей материального производства, в вопросах проектов создания программного обеспечения недопустимы простые подходы, основанные на умножении трудоемкости на среднюю производительность труда. Это вызвано, прежде всего, тем, что экономические показатели проекта нелинейно зависят от объема работ, а при вычислении трудоемкости допускается большая погрешность.

Поэтому для решения этой задачи используются комплексные и достаточно сложные методики, которые требуют высокой ответственности в применении и определенного времени на адаптацию.

Современные комплексные системы оценки характеристик проектов создания программного обеспечения могут быть использованы для решения следующих задач:

- предварительная, постоянная и итоговая оценка экономических параметров проекта: трудоемкость, длительность, стоимость;
- оценка рисков по проекту: риск нарушения сроков и невыполнения проекта, риск увеличения трудоемкости на этапах отладки и сопровождения проекта и пр.;
- принятие оперативных управленческих решений – на основе отслеживания определенных метрик проекта можно своевременно предупредить возникновение нежелательных ситуаций и устранить последствия непродуманных проектных решений.

Одним из ключевых способов повышения качества программного обеспечения, снижения трудозатрат и рисков является анализ сложности разрабатываемого программного обеспечения при помощи разнообразных метрик.

В современных условиях большинство программных проектов создается на основе объектно-ориентированного подхода, в связи с чем существует значительное количество метрик, позволяющих получить оценку сложности объектно-ориентированных проектов.

Взвешенная насыщенность класса 1 (Weighted Methods Per Class (WMC))

Отражает относительную меру сложности класса на основе цикломатической сложности каждого его метода. Класс с более сложными методами и большим количеством методов считается более сложным. При вычислении метрики родительские классы не учитываются.

Взвешенная насыщенность класса 2 (Weighted Methods Per Class (WMC2))

Мера сложности класса, основанная на том, что класс с большим числом методов, является более сложным, и что метод с большим количеством параметров также является более сложным. При вычислении метрики родительские классы не учитываются.

Глубина дерева наследования (Depth of inheritance tree)

Длина самого длинного пути наследования, заканчивающегося на данном модуле. Чем глубже дерево наследования модуля, тем может оказаться сложнее предсказать его поведение. С другой стороны, увеличение глубины даёт больший потенциал повторного использования данным модулем поведения, определённого для классов-предков.

Количество детей (Number of children)

Число модулей, непосредственно наследующих данный модуль. Большие значения этой метрики указывают на широкие возможности повторного использования; при этом слишком большое значение может свидетельствовать о плохо выбранной абстракции.

Связность объектов (Coupling between objects)

Количество модулей, связанных с данным модулем в роли клиента или поставщика. Чрезмерная связность говорит о слабости модульной инкапсуляции и может препятствовать повторному использованию кода.

Отклик на класс (Response For Class)

Количество методов, которые могут вызываться экземплярами класса; вычисляется как сумма количества локальных методов, так и количества удаленных методов

Отсутствие сцепления в методах (LCOM).

Отсутствие сцепления в методах (Lack Cohesion Of Methods-LCOM) позволяет оценить зависимость методов класса друг от друга. Для вычисления этой метрики подсчитывается количество пар методов, которые не используют общие атрибуты класса. Затем подсчитывается количество пар методов, которые используют общие переменные. Метрика LCOM равна разности между первым числом и вторым. Если при этом получается отрицательное число, то значение метрики считается равным 0.

Количество строк текста на класс (KLOC).

Количество строк текста на класс в килобайтах (Lines Of Class in K-KLOC) позволяет оценить сложность реализации класса.

Количество унаследованных методов (NMI).

Количество методов, унаследованных от предков (Numer of Methods Inherited - NMI) позволяет классу от его предков. В отличие от метрики DIT, эта метрика оценивает не количество уровней в иерархии наследования, а именно зависимость от предков. Возможно базовые классы не оказывают сильного влияния на потомков, если имеют небольшое количество методов.

Количество потенциально наследуемых методов (PMI).

Количество методов, унаследованных классом от предков (Potencial Methods Inherited - PMI) + количество методов в самом классе. С помощью этой метрики вычисляется степень функциональности, которую данный класс предоставляет своим потомкам.

Степень методологической сцепленности (DMC).

Степень методологического сцепления (Density of Methodological Cohensiveness - DMC). Эта метрика вычисляется как отношение количества пар методов, которые используют общие атрибуты класса, к общему числу пар методов. В отличие от метрики LCOM, эта метрика вычисляется более просто.

Постоянный анализ сложности разрабатываемых систем основанный на данных метриках, позволяет избежать рисков чрезмерного усложнения их структуры, что непосредственно сказывается на сложности разработки, поддержки и внедрения.